

Usulan Penjadwalan dengan menggunakan Algoritma Non delay untuk Mengurangi Makespan Di CV.X

Muhammad Fikri*, Endang Prasetyaningsih, Reni Amaranti

Prodi Teknik Industri, Fakultas Teknik, Universitas Islam Bandung, Indonesia.

*fmuh98@gmail.com, endangpras@gmail.com, amarantireni@gmail.com

Abstract. CV. X is a manufacturing company engaged in the production of molding and die casting, as well as machining services. The company produces two main types of products: molding and die casting. CV. X implements make-to-order (MTO) and engineering-to-order (ETO) market response systems. Currently, the company uses a first-come-first-served (FCFS) scheduling system to determine the production sequence. This system causes delays in product completion. This study proposes a non-delay algorithm as a solution to improve production scheduling efficiency. This algorithm ensures that no machine is idle when there is an operation that requires it. Simulation results show that the non-delay algorithm can reduce waiting time and mean flowtime, which ultimately reduces makespan. Compared to the FCFS system, makespan can be reduced by up to 8% from 24,468 minutes to 22,560 minutes. This study also proposes a procedure for creating schedules with a non-delay algorithm to help companies minimize delays in product completion.

Keyword : Scheduling, Non delay Algorithm, Makespan

Abstrak. CV. X merupakan sebuah perusahaan yang bergerak di bidang pembuatan moulding dan *Die Casting*, dan jasa *machining*. Terdapat 2 jenis produk yang dihasilkan CV. X yaitu produk Moulding dan Die Casting. Perusahaan menerapkan respon pasar *make to order* (MTO) dan *engineering to order* (ETO). Perusahaan dalam menentukan penjadwalan produksin menggunakan prioritas *first come first serve* (FCFS) atau pesanan yang masuk terlebih dahulu akan dijadwalkan paling awal untuk dilakukan produksi. Dilihat dari kondisi saat ini, permasalahan yang dihadapi oleh perusahaan yaitu terjadinya keterlambatan dalam menyelesaikan produksi. Hal ini disebabkan karena perusahaan dalam menentukan penjadwalan produksi menggunakan prioritas *first come first serve* (FCFS). Algoritma Non delay dipilih menjadi usulan perbaikan dalam melakukan penjadwalan. Penjadwalan Algoritma Non delay ini tidak membiarkan satupun mesin menganggur jika pada saat yang sama terdapat operasi yang memerlukan mesin tersebut. Setelah dilakukan dengan penjadwalan ini waktu tunggu dan *mean flowtime* berkurang sehingga *makespan* dapat dikurangi. Berdasarkan penjadwalan dengan menggunakan metode usulan makespan dapat dikurangi sebesar 8% dari 24.468 menit menjadi 22.560 menit. Pada bagian akhir penelitian ini diusulkan prosedur membuat jadwal dengan Algoritma Non delay agar perusahaan dapat mengurangi makespan sehingga kemungkinan terlambat dalam menyelesaikan produk dapat berkurang.

Kata Kunci: Penjadwalan, Algoritma *Non delay*, *Makespan*.

A. Pendahuluan

Persaingan bisnis khususnya di sektor industri manufaktur di Indonesia terus berubah dan semakin kompetitif dari waktu ke waktu. Persaingan yang ketat memaksa industri harus berupaya meningkatkan daya saingnya. Untuk meningkatkan daya saing, suatu industri dituntut mampu memenuhi keinginan pelanggan. Dalam usaha untuk memenuhi keinginan dari pelanggan, suatu industri dituntut untuk melakukan aktivitas yang dapat memberikan kepuasan terhadap konsumen, salah satunya yaitu dengan menyelesaikan dan mengirimkan produk secara tepat waktu.

Salah satu cara untuk menyelesaikan dan mengirimkan produk secara tepat waktu yaitu melakukan penjadwalan produksi. Penjadwalan produksi merupakan bagian penting dari suatu perusahaan untuk menentukan kapan produk dapat diselesaikan dan dikirim kepada konsumen. Penjadwalan produksi yang baik dapat membantu perusahaan dalam mengatur dan mengelola produksi yang efisien sehingga dapat menghasilkan produk yang berkualitas dengan waktu produksi yang lebih cepat dan biaya yang lebih rendah.

Penjadwalan yang baik memiliki beberapa manfaat seperti perusahaan dapat menggunakan asetnya dengan efektif, mampu memberikan waktu pengiriman yang tepat waktu maupun lebih cepat. Fungsi utama penjadwalan adalah untuk membuat proses produksi berjalan lancar sesuai dengan waktu yang telah ditentukan, sehingga produk dapat diselesaikan dan dikirim secara tepat waktu [1].

Penjadwalan perusahaan saat ini yaitu dengan mendahulukan produksi dengan order yang datang lebih awal atau disebut juga *first come first serve* (FCFS). Namun penggunaan prioritas FCFS untuk penjadwalan produksi mengakibatkan beberapa produk terlambat untuk diselesaikan. Dilihat dari kondisi saat ini, permasalahan yang dihadapi oleh perusahaan yaitu perusahaan dalam menentukan produksinya selalu menggunakan prioritas *first come first serve* (FCFS) atau pesanan yang masuk terlebih dahulu akan dijadwalkan paling awal untuk dilakukan produksi. Penggunaan prioritas FCFS ini menjadi kurang efektif pada kasus apabila order lebih dari satu datang secara bersamaan, karena ada beberapa *job* harus di proses di mesin yang sama yang memiliki waktu proses lebih cepat dibandingkan *job* pertama, sehingga menimbulkan antrian lebih lama pada mesin. Sementara itu ada jenis mesin lain yang seharusnya dapat digunakan dalam kondisi menganggur sehingga menyebabkan delay. Selain itu pola aliran proses produksi pada tiap *job* memiliki aliran proses yang berbeda-beda, atau disebut juga dengan pola aliran *jobshop* [2].

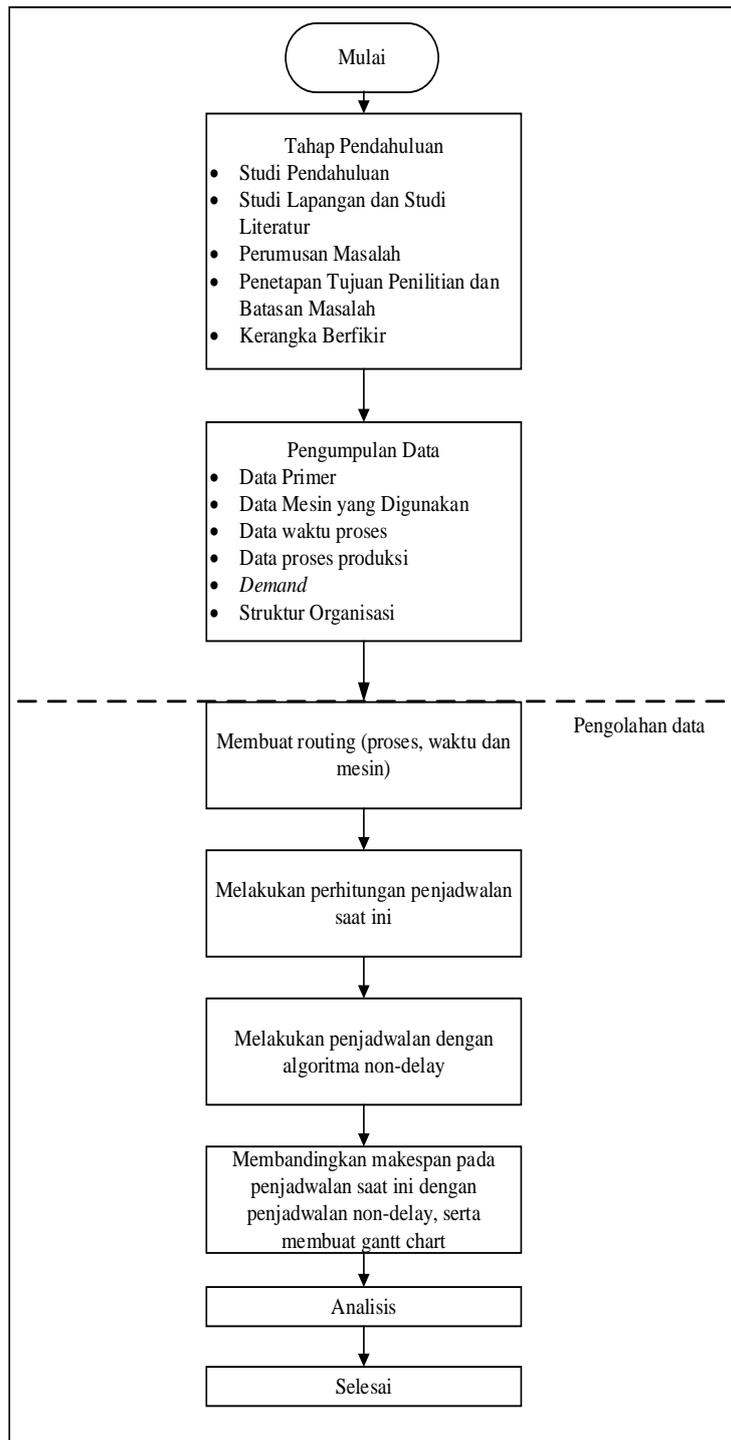
Hal ini membuat perusahaan kesulitan untuk menentukan prioritas *job* yang mana yang akan lebih dulu dikerjakan agar dapat diselesaikan sebelum *due date* yang telah ditentukan. Maka dari itu perlu dilakukan perbaikan dalam penjadwalan untuk menentukan *job* mana yang lebih dulu akan dikerjakan pada setiap mesinnya untuk mengurangi *makespan*. Dalam produksi aliran *jobshop* ada beberapa metode penjadwalan yang dapat digunakan, salah satunya yaitu penjadwalan mesin menggunakan Algoritma *Non delay*. Maka dari itu perusahaan harus melakukan upaya perbaikan dalam melakukan penjadwalan untuk mengurangi *makespan* dengan ekspektasi dapat mengurangi keterlambatan.

Berdasarkan latar belakang masalah yang ada di CV. X, tujuan penelitian meliputi:

1. Menentukan *makespan order* dengan prosedur penjadwalan saat ini.
2. Melakukan penjadwalan dengan Algoritma *Non delay* untuk mengurangi *makespan*.

B. Metodologi Penelitian

Penelitian ini berfokus membuat penjadwalan produksi *job shop* untuk mengurangi *makespan*. Penelitian dilakukan dengan terlebih dahulu mengidentifikasi penjadwalan saat ini. Identifikasi penjadwalan saat ini dilakukan untuk mengetahui *makespan*, waktu tunggu, *flowtime*, dan *mean flowtime* akan dijadikan sebagai perbandingan dengan penjadwalan yang akan diusulkan. Gambar 1 adalah Langkah penelitian yang dilakukan



Matriks Routing dan Matriks Waktu Untuk Persoalan Jobshop

Dalam penjadwalan *jobshop* diperlukan input berupa jumlah *job*, jumlah operasi dalam tiap *job* dan urutan operasi beserta mesin yang memprosesnya (*routing*). Hal ini ditampilkan dalam bentuk matriks waktu proses yang menyatakan urutan mesin yang memproses tiap-tiap urutan operasi. Suatu penjadwalan digambarkan dengan susunan balok-balok, dimana setiap balok merupakan *triplet* dari *job*-operasi-mesin. Panjang balok menyatakan waktu proses pekerjaan yang bersangkutan. Notasi triplet yang digunakan tiap balok (i, j, k) dimana i menunjukkan nomor pekerjaan, j menyatakan urutan pekerjaan dan k mewakili mesin yang diperlukan. Tabel matriks *routing* dan matriks waktu terdapat Pada Tabel 1 dan Tabel 2.

Tabel 1 Matriks Waktu Operasi

Job	Operasi		
	1	2	3
1	3	2	4
2	4	4	1
3	2	2	3
4	3	3	3

Keterangan :

$Q_{i,j,k}$: Job i , waktu proses operasi j , diproses pada mesin k

$Q_{2,1,1}$: Job 2, waktu proses operasi 1, diproses pada mesin 1

Tabel 2 Matriks Routing

Job	Operasi		
	1	2	3
1	M2	M3	M1
2	M1	M3	M2
3	M2	M4	M3
4	M3	M1	M2

Keterangan :

$Q_{i,j,k}$: Job i , operasi j , diproses pada mesin k

$Q_{2,1,1}$: Job 2, operasi 1, diproses pada mesin 1

Penjadwalan Non delay

Penjadwalan yang diusulkan yaitu dilakukan dengan pendekatan Algoritma *non delay*. Penjadwalan *Non delay* adalah metode penjadwalan aktif yang tidak membiarkan mesin menjadi idle bila suatu operasi dapat dimulai [4]. Pada umumnya penjadwalan *jobshop* dengan Algoritma *Non delay* hanya menggunakan satu mesin (mesin tunggal) pada setiap prosesnya, yaitu dengan Algoritma sebagai berikut:

Langkah 1: $t = 0$, $PS_t = 0$ (yaitu jadwal parsial yang mengandung t operasi terjadwal). Set S_t (yaitu kumpulan operasi yang siap dijadwalkan) sama dengan seluruh operasi tanpa pendahulu.

Langkah 2: Tentukan $c^* = \min(c_j)$ dimana c_j adalah saat paling awal operasi j dapat mulai dikerjakan. Tentukan pula m^* , yaitu mesin di mana c^* dapat direalisasi. Melihat apakah *ready time* mesin minimum lebih dari satu. Jika ya, berarti lanjutkan ke langkah 3. Jika *time* mesin minimum lebih dari 1, pilihlah operasi berdasarkan aturan prioritas berdasarkan *Short Processing Time* (SPT) atau waktu proses tercepat. Jika masih ada lebih dari satu operasi yang sama berdasarkan SPT maka prioritas selanjutnya pilihlah operasi berdasarkan *Most Work Remaining* (MWKR) atau jumlah *job* terbanyak yang belum dikerjakan. Jika setelah prioritas MWKR masih terdapat lebih dari satu operasi yang dapat dijadwalkan, maka pilih secara *random*.

Langkah 3: Untuk setiap operasi dalam PS_t yang memerlukan mesin m^* dan memiliki $c_j = c^*$ buat suatu aturan prioritas tertentu. Tambahkan operasi yang prioritasnya paling besar ke dalam PS_t sehingga terbentuk suatu jadwal parsial untuk tahap berikutnya.

Langkah 4: Buat suatu jadwal parsial baru PS_{t+1} dan memperbaiki kumpulan data dengan cara: 1. Menghilangkan operasi j dari S_t , 2. Buat S_{t+1} dengan cara menambah pengikut langsung operasi j yang telah dihilangkan, serta 3. menambahkan satu (+1) pada t .

Langkah 5: Kembali ke langkah 2 sampai seluruh pekerjaan terjadwalkan.

Performansi Penjadwalan

Performansi penjadwalan digunakan untuk menentukan metode mana yang lebih baik untuk diterapkan pada perusahaan. Parameter performansi yang dapat digunakan diantaranya adalah [5]:

1. *Efficiency Index* (EI): *Efficiency Index* adalah perbandingan antara metode usulan dengan metode yang digunakan perusahaan.

$$EI = \frac{\text{Makespan perusahaan}}{\text{Makespan usulan}} \dots\dots\dots(I)$$

Apabila $E_i = 1$ dapat disimpulkan bahwa kedua metode memiliki performansi yang sama, jika nilai $E_i < 1$ maka performansi metode usulan dapat dikatakan kurang baik, begitupun sebaliknya, jika nilai $E_i > 1$ maka metode usulan dapat dikatakan performansinya lebih baik daripada metode saat ini.

2. *Relative Error (RE)*

Relative Error dapat digunakan untuk mengetahui seberapa jauh perbedaan *makespan* yang dihasilkan oleh kedua metode. Untuk mengetahui perbedaan *makespan* dengan *relative error* dapat digunakan persamaan:

$$RE = \frac{\text{Makespan perusahaan} - \text{makespan usulan}}{\text{Makespan perusahaan}} \times 100\% \dots\dots\dots(II)$$

C. Hasil Penelitian dan Pembahasan

Pengolahan data dimulai dengan menguraikan pekerjaan berdasarkan indeks (*i, j, k*) yang menunjukkan bahwa pekerjaan *i* dengan operasi *j* pada mesin *k* seperti yang ditunjukkan Tabel 23 dan Tabel 4.

Tabel 3. Matriks Routing

No	Kode Job	Kode Komponen	Operasi (Mesin)						
			O-1	O-2	O-3	O-4	O-5	O-6	O-7
1	J1	LRMP	M3	M1	M8	M10			
2	J2	EJMP	M3	M1	M8	M10			
3	J3	GPMP	M3	M1	M8	M10			
4	J4	GBMP	M3	M1	M8	M10			
5	J5	CPMP	M1	M2	M5	M8	M10		
6	J6	RPMP	M1	M2	M5	M8	M10		
7	J7	SPMP	M1	M2	M5	M8	M10		
8	J8	EPMP	M1	M2	M5	M8	M10		
9	J9	TPMP	M1	M2	M5	M8	M10		
10	J10	BPMP	M1	M2	M5	M8	M10		
11	J11	LRMR	M3	M1	M8	M10			
12	J12	EJMR	M3	M1	M8	M10			
13	J13	GPMR	M3	M1	M8	M10			
14	J14	GBMR	M3	M1	M8	M10			
15	J15	CPMR	M1	M2	M5	M8	M10		
16	J16	RPMR	M1	M2	M5	M8	M10		
17	J17	SPMR	M1	M2	M5	M8	M10		
18	J18	EPMR	M1	M2	M5	M8	M10		
19	J19	TPMR	M1	M2	M5	M8	M10		
20	J20	BPMR	M1	M2	M5	M8	M10		
21	J21	COPMB	M1	M5	M4	M2	M10		
22	J22	RPMB	M1	M5	M4	M2	M10		
23	J23	CPMB	M5	M1	M10				
24	J24	EJMB	M1	M2	M5	M4	M2	M10	
25	J25	COPMI	M1	M5	M4	M2	M10		
26	J26	RPMI	M1	M2	M5	M4	M2	M10	
27	J27	CPMI	M1	M2	M1	M5	M4	M2	M10
28	J28	EJMI	M4	M1	M10				
29	J29	COPMIK	M1	M5	M1	M10			

No	Kode Job	Kode Komponen	Operasi (Mesin)						
			O-1	O-2	O-3	O-4	O-5	O-6	O-7
30	J30	RPMIK	M1	M2	M5	M4	M2	M10	
31	J31	CPMIK	M4	M1	M10				
32	J32	EJMIK	M1	M2	M1	M5	M2	M10	

Tabel 4. Matriks Waktu

No	Kode Job	Kode Komponen	Operasi (Menit)							
			O-1	O-2	O-3	O-4	O-5	O-6	O-7	
1	J1	LRMP	30	26	20	8				
2	J2	EJMP	264	252	240	96				
3	J3	GPMP	104	92	84	32				
4	J4	GBMP	108	92	80	32				
5	J5	CPMP	31	22	1.808	21	9			
6	J6	RPMP	27	19	1.827	22	9			
7	J7	SPMP	26	21	1.801	19	9			
8	J8	EPMP	26	19	1.830	20	9			
9	J9	TPMP	23	19	1.810	20	9			
10	J10	BPMP	26	20	1.800	20	9			
11	J11	LRMR	29	25	21	8				
12	J12	EJMR	308	294	280	112				
13	J13	GPMP	108	84	76	32				
14	J14	GBMR	124	100	80	32				
15	J15	CPMR	31	22	1.171	21	9			
16	J16	RPMR	27	19	1.230	22	9			
17	J17	SPMR	27	21	1.151	19	9			
18	J18	EPMR	28	19	1.143	20	9			
19	J19	TPMR	25	19	1.246	20	9			
20	J20	BPMR	26	21	1.174	19	9			
21	J21	COPMB	55	615	435	45	9			
22	J22	RPMB	81	465	145	35	9			
23	J23	CPMB	440	75	9					
24	J24	EJMB	78	28	315	165	30	8		
25	J25	COPMI	77	465	155	35	9			
26	J26	RPMI	73	28	315	165	45	8		
27	J27	CPMI	79	30	60	315	165	53	8	
28	J28	EJMI	265	45	10					
29	J29	COPMIK	45	260	31	9				
30	J30	RPMIK	84	28	315	165	45	8		
31	J31	CPMIK	265	555	10					
32	J32	EJMIK	77	30	60	315	27	8		

Penjadwalan Saat ini

Penjadwalan perusahaan saat ini dalam menentukan proses produksi yaitu dengan mendahulukan *order* yang masuk paling awal atau bisa disebut dengan *first come first serve* (FCFS). Pada penelitian ini, diterapkan beberapa asumsi seperti seluruh *job* diasumsikan datang secara bersamaan, operator dan mesin selalu tersedia, mesin dalam keadaan *ready* dan langsung diproses

pada mesin. Berikut merupakan contoh perhitungan *makespan* awal pengerjaan seluruh *job* berdasarkan penjadwalan perusahaan saat ini.

Job 1 operasi 1

Mesin = 3
 Waktu Proses = 30 menit
 Waktu Mulai = Waktu selesai pada operasi terakhir
 = 0
 Waktu selesai = waktu proses + waktu mulai
 = 0 + 30 menit = 30 menit
 Waktu tunggu = waktu mulai – waktu selesai operasi sebelumnya
 = 0 – 0 = 0
Flowtime = waktu proses + waktu tunggu
 = 30 + 0 = 30 menit

Job 1 operasi 2

Mesin = 1
 Waktu Proses = 26 menit
 Waktu Mulai = Waktu selesai pada operasi terakhir yaitu operasi 1
 = 30 menit
 Waktu selesai = waktu proses + waktu mulai
 = 26 + 30 menit = 56 menit
 Waktu tunggu = waktu mulai – waktu selesai operasi sebelumnya
 = 30 – 30 = 0 menit
Flowtime = waktu proses + waktu tunggu
 = 26 + 0 = 26 menit

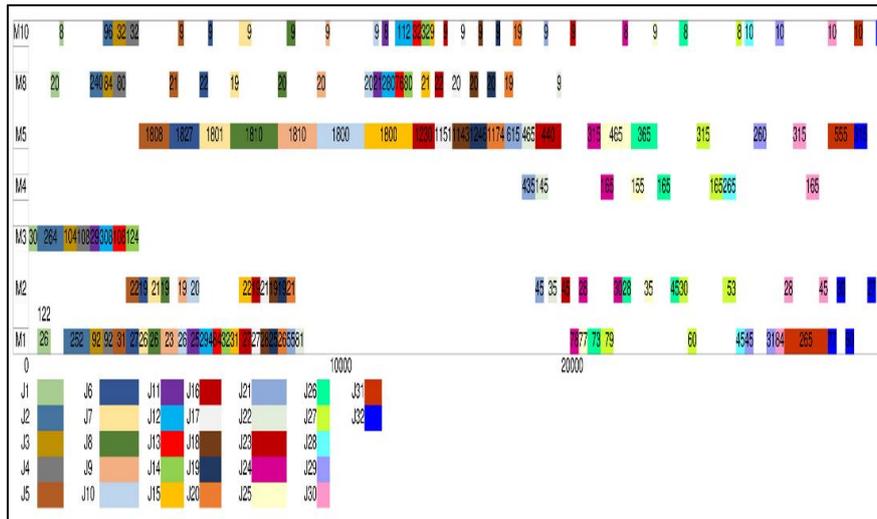
Job 1 operasi 3

Mesin = 8
 Waktu Proses = 20 menit
 Waktu Mulai = Waktu selesai pada operasi terakhir yaitu operasi 2
 = 56 menit
 Waktu selesai = waktu proses + waktu mulai
 = 20 + 56 menit = 76 menit
 Waktu tunggu = waktu mulai – waktu selesai operasi sebelumnya
 = 56 – 56
 = 0 menit
Flowtime = waktu proses + waktu tunggu
 = 20 + 0 = 20 menit

Job 1 operasi 4

Mesin = 10
 Waktu Proses = 8 menit
 Waktu Mulai = Waktu selesai pada operasi terakhir yaitu operasi 2
 = 76 menit
 Waktu selesai = waktu proses + waktu mulai
 = 8 + 76 menit = 84 menit
 Waktu tunggu = waktu mulai – waktu selesai operasi sebelumnya
 = 76 – 76 = 0 menit
Flowtime = waktu proses + waktu tunggu
 = 8 + 0 = 8 menit

Kemudian setelah perhitungan selesai, maka dibuatkan *ganttt chart* berdasarakan penjadwalan saat ini. Gambar 1 adalah *ganttt chart* penjadwalan saat ini.



Gambar 1. Penjadwalan saat ini

Berikut merupakan rekapitulasi hasil perhitungan saat ini dengan menggunakan penjadwalan saat ini yang ditunjukkan pada Tabel 5

Tabel 5. Rekapitulasi nilai makspan, waktu tunggu, flowtime, dan mean flowtime

Makespan (menit)	F _{total} (menit)	F _{mean} (menit)	Waktu tunggu (menit)
24.460	477.397	149.19	447.434

Penjadwalan Algoritma Non delay

Penjadwalan Algoritma *Non delay* merupakan kumpulan jadwal *feasible* dimana tidak ada satupun mesin dibiarkan menganggur dengan melihat *ready time job* yang terkecil. Apabila *ready time job* terkecil lebih dari satu maka operasi dipilih berdasarkan waktu proses tercepat. Apabila terdapat lebih dari satu operasi terkecil maka pemilihan operasi berdasarkan jumlah *job* terbanyak yang belum dikerjakan. Jika masih terdapat lebih dari satu operasi maka selanjutnya dipilih secara random.

- Langkah 1: $t = 0, P_{S_t} = 0$ (yaitu jadwal parial yang mengandung t operasi terjadwal). Set S_t (yaitu kumpulan operasi yang siap dijadwalkan) sama dengan seluruh operasi tanpa pendahulu.
- Langkah 2: Tentukan $c^* = \min (c_j)$ dimana c_j adalah saat paling awal operasi j dapat mulai dikerjakan. Tentukan pula m^* , yaitu mesin di mana c^* dapat direalisasikan.
- Langkah 3: Untuk setiap operasi dalam P_{S_t} yang memerlukan mesin m^* dan memiliki $c_j = c^*$ buat suatu aturan prioritas tertentu. Tambahkan operasi yang prioritasnya paling besar ke dalam P_{S_t} sehingga terbentuk suatu jadwal parsial untuk tahap berikutnya.
- Langkah 4: Buat suatu jadwal parsial baru P_{t+1} dan memperbaiki kumpulan data dengan cara: 1. Menghilangkan operasi j dari S_t 2. Buat S_{t+1} dengan cara menambah pengikut langsung operasi j yang telah dihilangkan,serta 3. Menambahkan satu pada t .
- Langkah 5: Kembali ke langkah 2 sampai seluruh pekerjaan terjadwalkan.

Kemudian setelah dilakukan penjadwalan, maka dibuat gantt chart. Berikut merupakan contoh perhitungan penjadwalan menggunakan Algoritma *non delay*:

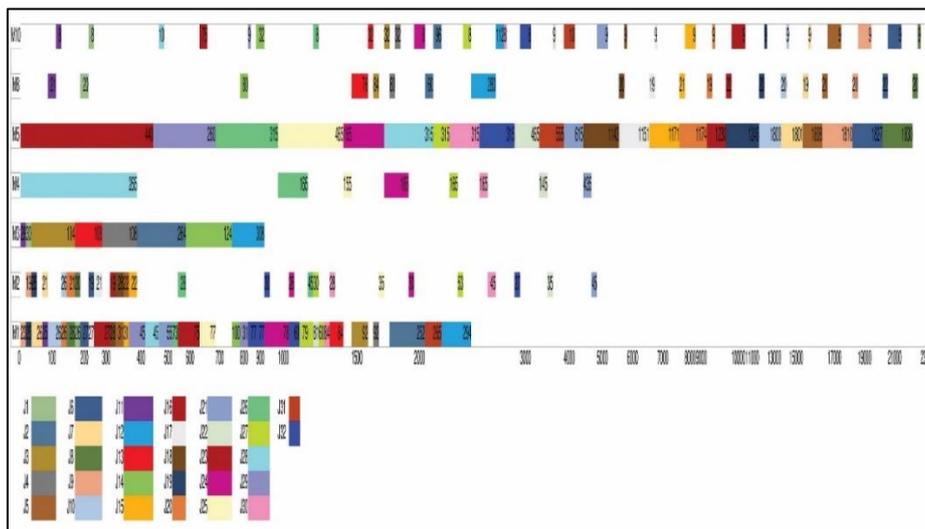
Langkah 1: Semua pekerjaan dimulai pada $t = 0$, karena belum ada proses yang dijadwalkan. Menentukan *job*, operasi dan mesin yang akan dijadwalkan pada P_{S_t} , yang dijadwalkan adalah {113, 213, 313, 413, 511, 611, 711, 811, 911, 1011, 1113, 1213, 1313, 1413, 1511, 1611, 1711, 1811, 1911, 2011, 2111, 2211, 2315, 2411, 2511, 2611, 2711,2814, 2911, 3011, 3111, 3211}. Dari *job* yang dijadwalkan tentukan waktu mulai (C_j) dan waktu proses (t_{ij}) sehingga diketahui waktu penyelesaiannya (r_j).

- Langkah 2: $c_{113} = 0, c_{213} = 0, c_{313} = 0, c_{413} = 0, c_{511} = 0, c_{611} = 0, c_{711} = 0, c_{811} = 0, c_{911} = 0, c_{1011} = 0, c_{1113} = 0, c_{1213} = 0, c_{1313} = 0, c_{1413} = 0, c_{1511} = 0, c_{1611} = 0, c_{1711} = 0, c_{1811} = 0, c_{1911} = 0, c_{2011} = 0, c_{2111} = 0, c_{2211} = 0, c_{2315} = 0, c_{2411} = 0, c_{2511} = 0, c_{2611} = 0, c_{2711} = 0, c_{2814} = 0, c_{2911} = 0, c_{3011} = 0, c_{3111} = 0, c_{3211} = 0, c^* = 0, m^* = 1, 3, 4, 5.$
- Langkah 3: Operasi yang memerlukan $m^* = 1$ adalah 511, 611, 711, 811, 911, 1011, 1511, 1611, 1711, 1811, 1911, 2011, 2111, 2211, 2411, 2511, 2611, 2711, 2911, 3011, 3111, 3211. Operasi yang memerlukan $m^* = 3$ adalah 113, 213, 313, 413, 1113, 1213, 1313, 1413. Operasi yang memerlukan $m^* = 4$ adalah 2814. Operasi yang memerlukan $m^* = 5$ adalah 2315. Dengan aturan SPT, maka yang terpilih adalah 911, 1113, 2315, 2814 untuk digabungkan dengan P_{S_i} . $P_{S_i} = \{911, 1113, 2315, 2814\}$
- Langkah 4: 911, 1113, 2315, 2814 dicoret dari S_i , tambahkan operasi baru yang merupakan pengikut langsung dari 911, 1113, 2315, 2814 yaitu 922, 1121, 2321, 2821.
- Langkah 5: Kembali ke langkah 2.

Tabel 6. Penjadwalan menggunakan Algoritma non delay stage 0

Stage	Mesin										St	Cj	Tj (Menit)	Rj (Menit)	c*	t*	Pst
	1	2	3	4	5	6	7	8	9	10							
0											1413	0	124	124			
											1511	0	31	31			
											1611	0	27	27			
											1711	0	27	27			
											1811	0	28	28			
											1911	0	25	25			
											2011	0	26	26			
											2111	0	55	55			
											2211	0	81	81			
											2315	0	440	440	5	2315	
											2411	0	78	78			
											2511	0	77	77			
											2611	0	73	73			
											2711	0	79	79			
											2814	0	265	265	4	2814	
											2911	0	45	45			
											3011	0	84	84			
											3111	0	265	265			
											3211	0	77	77			
1	23	0	29	265	440	0	0	0	0	0	113	29	30	59			
											213	29	264	293			
											313	29	104	133			
											413	29	108	137			
											511	23	31	54			
											611	23	27	50			
											711	23	26	49			
											811	23	26	49			
											922	83	19	83	2	922	
											1011	23	26	49			
											1121	29	25	54			
											1213	29	308	337			
											1313	29	108	137			

Gambar 2. adalah gantt chart dengan menggunakan penjadwalan non delay



Gambar 3. Gantt chart penjadwalan dengan Algoritma non delay

Berikut adalah rekapitulasi perhitungan dengan menggunakan jadwal Algoritma non delay yang terdapat pada Tabel 7.

Tabel 7. Rekapitulasi Makespan, Waktu tunggu, Flowtime, dan mean flowtime

Makespan (menit)	F _{total} (menit)	F _{mean} (menit)	Waktu tunggu (menit)
22.560	201.880	4.038	169.188

Setelah pengolahan data yang telah dilakukan, selanjutnya yaitu menghitung performansi penjadwalan, perhitungan ini bertujuan untuk mengetahui dan menentukan metode penjadwalan mana yang lebih baik, pada perhitungan performansi penjadwalan, dilakukan dua perhitungan yaitu menghitung *efisiensi index* dan *relative error*. Perhitungan performansi dapat dihitung dengan menggunakan persamaan (I) sebagai berikut:

$$EI = \frac{\text{Makespan perusahaan}}{\text{Makespan usulan}}$$

$$EI = \frac{24468}{22560} = 1,08$$

Nilai dari *relative error* dihitung dengan persamaan (II) sebagai berikut:

$$RE = \frac{\text{Makespan perusahaan} - \text{makespan usulan}}{\text{Makespan perusahaan}} \times 100\%$$

$$RE = \frac{24468 - 22560}{24168} \times 100\% = 8\%$$

Berdasarkan perhitungan EI, diketahui nilai perhitungan EI yaitu 1,08 atau > 1, jika nilai EI > 1 maka penjadwalan usulan dinilai memiliki performansi yang lebih baik. Selain itu dengan perhitungan menggunakan Algoritma *Non delay*, *makespan* berhasil dikurangi sebesar 8% atau dikurangi sebesar 1900 menit dari penjadwalan awal perusahaan. merupakan rekapitulasi perbedaan *makespan*, waktu tunggu, *flowtime* yang terdapat pada Tabel 8

Tabel 8. Rekapitulasi Makespan, Waktu tunggu, dan Flowtime

Kondisi	Makespan	F _{total} (menit)	F _{mean} (menit)	Waktu tunggu (menit)
Saat ini	24.460	477.397	14918.65	4474.34
Setelah perbaikan	22.560	201.880	6305.22	1691.88

D. Kesimpulan

Kesimpulan yang diperoleh berdasarkan hasil penelitian yaitu penjadwalan usulan terbukti dapat mengurangi *makespan*. Selain itu nilai waktu tunggu, *flowtime*, dan *mean flowtime* dapat dikurangi, artinya pekerjaan yang berada pada dalam sistem dapat berkurang dan meningkatkan utilitas mesin sehingga dapat memproses lebih banyak pekerjaan. Nilai *efficiency index* 1,08 atau > 1 yang artinya penjadwalan usulan lebih baik dibandingkan penjadwalan saat ini. Perhitungan *relative error* menunjukkan bahwa *makespan* dapat dikurangi 8%.

Acknowledge

Saya ucapkan terima kasih kepada Ibu Dr. Endang Prasetyaningsih, Ir., MT. dan Ibu Dr. Ir. Reni Amaranti, ST., MT., IPM. yang telah bersedia meluangkan waktu untuk memberikan bimbingan dan arahan selama penyusunan penelitian ini. Tak lupa, ucapan terima kasih juga ditujukan kepada seluruh pihak di perusahaan yang telah memberikan izin dan membantu penulis dalam melakukan penelitian ini.

Daftar Pustaka

- [1] Mulya, M. F., Trisanto, D., dan Rismawati, N. (2020). Analisis dan Implementasi Metode Earliest Due Date (EDD) untuk Meminimalisir Keterlambatan dalam Proses Penjadwalan Perbaikan Kendaraan. *Faktor Exacta*, 13(3), 168-175.
- [2] Nur'aeni, R., Muhammad, C., dan Amaranti, R. (2022). Penjadwalan Mesin Menggunakan Algoritma *Non Delay* untuk Mereduksi *Mean Tardiness* pada Lingkungan Batch Production. *Bandung Conference Series: Industrial Engineering Science*, 2(1),

- 122-132.
- [3] Bedworth, D.D., dan Bailey, J.E. (1987). *Integrated Production Control System: Management, Analysis, Design*. New York: John Wiley dan Sons, Inc.
 - [4] Fithri, P., dan Ramawinta, F. (2013). Penjadwalan Mesin Dengan Menggunakan Algoritma Pembangkitan Jadwal Aktif dan Algoritma Penjadwalan *Non-Delay* Untuk Produk Hydrotiller dan Hammermil Pada CV. Cherry Sarana Argo. *Optimasi Sistem Industri*, 12(2), 377-399.
 - [5] Ginting, R. (2009). *Penjadwalan Mesin*. Yogyakarta: Graha Ilmu.